NAME: ______________________________

SPIRE ID: ______________________________

COMPSCI 250
Introduction to Computation
Second Midterm Fall 2023

D. A. M. Barrington and G. Parvini 9 November 2023

DIRECTIONS:

- Answer the problems on the exam pages.
- There are four problems on pages 2-8, some with multiple parts, for 100 total points plus 5 extra credit. Final scale will be determined after the exam.
- Page 9 contains useful definitions and is given to you separately – do not put answers on it!
- If you need extra space use the back of a page –both sides are scanned.
- If absolutely necessary, you may attach additional pages to be graded, but this is a big nuisance to us.
- No books, notes, calculators, or collaboration.
- Scrap paper during the exam may be used, if it starts out blank. Please transfer any answers to the exam pages.
- You may say "pass" for any question or lettered part of a question, except for the true/false and the extra credit question, and receive 20% of the points.

1	/10
2	/15
3	/15
4	/40+5
5	/20
Total	/100+5

Question 1 (10): (Induction 1) Define a sequence of integers by the rule $a_0 = 7$ and, for any natural n, $a_{n+1} = a_n + n - 3$.

Prove, by induction, that for any natural n, $a_n = \frac{1}{2}(n^2 - 7n + 14)$.

(**Hint:** This can be done with ordinary induction.)

Question 2 (15): (Induction 2) Prove, for any natural n, that the number $2^n 3^n$ divides the number $(3n)!$ (the factorial of $3n$).

Hint: Remember that $0! = 1$. You will probably want to compute the number $(3n+3)!/(3n)!$. Ordinary induction is sufficient.

Question 3 (15) (Induction 3) :

Let $\Sigma = \{a, b, c\}$. A **palindrome** is a string w whose reversal w^r is itself. For any natural n, let $pal(n)$ be the number of palindromes of length n over Σ.

Prove by induction for all *positive* naturals n that $pal(n) = 3^{(n+1)/2}$, where "/" here is floor division or Python "//". (So we are proving that $pal(n) = 3^{n/2}$ if n is even, and $pal(n) = 3^{(n+1)/2}$ if n is odd.)

(**Hint**: Determine the value of $pal(n+2)$ in terms of $pal(n)$, and complete the proof with either odd-even induction or strong induction.)

Question 4 (40+5): (Searches and Search Trees) This problem concerns an undirected graph U and a directed graph D, each of which has nine nodes and eleven edges. Both are copied on the supplemental sheet. We begin with U:

```
(a) =================== (b)
 |                       |
 |                       |
(c) =========== (d) === (e)
 |               |       |
 |               |       |
(f) === (g) === (h) === (i)
```

- (a, 10) Carry out a DFS search for the undirected graph U, beginning with node c and with no goal node, indicating which nodes go on and off of the open list in what order. When two or more nodes need to come off the stack, and they entered at the same time, take the one that comes earlier alphabetically.

  Draw the DFS tree, indicating the tree edges and the back edges.

```
(a) =================== (b)
 |                       |
 |                       |
(c) =========== (d) === (e)
 |               |       |
 |               |       |
(f) === (g) === (h) === (i)
```

- (b, 10) Carry out a BFS search for the undirected graph U, beginning with node c and with no goal node, indicating which nodes go on and off of the open list in what order. When two or more nodes need to come off the queue, and they entered at the same time, take the one that comes earlier alphabetically.

  Draw the BFS tree, indicating the tree edges and the non-tree edges.

Here is the directed graph D. It is obtained from U by assigning exactly one direction for each of the edges in U. The caret symbols under nodes a, b, and e indicate directed edges upward to those nodes. The v symbols indicate directed edges downward to c and g.

```
(a) <================== (b)
 ^                      ^

 |                      |
(c) ==========> (d) <== (e)
 |               |      ^
 v               v      |
(f) <== (g) <== (h) ==> (i)
```

- (c, 15) Conduct a DFS of the directed graph D, beginning with node c and with no goal node, indicating which nodes go on and off of the open list in what order. Again, nodes that entered the stack at the same time come off in alphabetical order.

  Draw the DFS tree, indicating each edge as a tree edge, back edge, cross edge, or forward edge.

```
(a) <================== (b)
 ^                      ^
 |                      |
(c) ==========> (d) <== (e)
 |               |       ^
 v               v       |
(f) <== (g) <== (h) ==> (i)
```

- (d, 5) The **strongly connected component** of a node u in a directed graph is the set of nodes v such that there are paths both from u to v and from v to u. (Thus the strongly connected components are the equivalence classes of the equivalence relation $P(u, v) \wedge P(v, u)$.) Determine *all* the strongly connected components in the directed graph D. Justify your answers, possibly referring to your answer in part (c).

- (e, 5XC) Conduct a BFS of the directed graph D, also beginning with node c. In this case it is sufficient to draw the tree with its tree and non-tree edges – we don't need the sequence of events.

Question 5 (20): The following are ten true/false questions, with no explanation needed or wanted, no partial credit for wrong answers, and no penalty for guessing.

- (a) An undirected graph with n nodes may have any integer number of edges from 0 through $(n^2 - n)/2$.

- (b) Let G be a directed graph such that every node has at least one outgoing edge. Then G must have at least one directed cycle, so it is not acyclic.

- (c) If n is a natural, and we know that $n\%5 = 4$ and $n\%7 = 2$, we may conclude that $n\%35 = 9$. (Here the % operator is the modular division operator, which in this case is the same in either Java or Python.)

- (d) If a, b, c, d, and e are any five strings, then the strings $a(bc)^R(de)^R$ and $ae^Rd^Rc^Rb^R$ are the same. (Here a^R, *e.g.*, is the reversal of the string a).

- (e) The **degree** of a node in an undirected graph is the number of its neighbors. If we add together the degrees of all the nodes in such an undirected graph, the result must be even.

- (f) Recall that an undirected graph is defined to be **bipartite** if its nodes can be partitioned into two sets, such that every edge has an endpoint in each set. Then the undirected graph U from Question 4 is *not* bipartite.

- (g) The operation $\rightarrow$ on boolean expressions is neither commutative nor associative.

- (h) Consider the boolean expression with infix representation "$(a \wedge b) \vee c$". Then this expression has prefix representation "$\vee \wedge abc$" and postfix representation "$abc \wedge \vee$".

- (i) Let G be a directed graph with finitely many nodes. Suppose we conduct a DFS or a BFS in such a way that once a node has left the open list, it can never be put on the open list again. Then both searches are guaranteed to terminate.

- (j) Let a and b be two naturals that are each less than or equal to the Fibonacci number $F(n)$. Then the Euclidean Algorithm, on inputs a and b, could not terminate with more than $n + 3$ divisions.

COMPSCI 250 Second Midterm Supplementary Handout: 9 November 2023

Here are copies of the undirected graph U and the directed graph D for Question 4. Each has nine nodes and eleven edges.

Undirected graph U:

```
(a) =================== (b)
 |                       |
 |                       |
(c) =========== (d) === (e)
 |               |       |
 |               |       |
(f) === (g) === (h) === (i)
```

Directed graph D:

```
(a) <================== (b)
 ^                       ^
 |                       |
(c) ==========> (d) <== (e)
 |               |       ^
 v               v       |
(f) <== (g) <== (h) ==> (i)
```